

Algorithmische Graphentheorie

5. Übungsblatt

Jasper Gude Pia Röttgers

23. Mai 2026

/ 25

Aufgabe 1 – Paarungen (Matchings) in Bäumen

- a) Ein Baum ist ein kreisfreier, zusammenhängender Graph. Ein Baum besitzt ein perfektes Matching genau dann, wenn die Anzahl der Knoten gerade ist und jedes Blatt keine Geschwisterknoten hat.

Die Anzahl der Knoten muss gerade sein, da in einem perfekten Matching jeder Knoten mit einem anderen gematcht wird.

Jedes Blatt muss Einzelblatt sein, da der Elternknoten nur mit einem Blatt gematcht werden kann.

Also ist ein perfektes Matching im Baum eindeutig, da jedes Blatt mit seinem Elternknoten gematcht werden muss. Die Kante ist sicher im Matching enthalten und kann aus dem Graphen entfernt werden. So entsteht ein neues Blatt, für das die selbe Regel gilt.

/ 3

- b) Der Algorithmus 1 berechnet ein größtes Matching.

Der Algorithmus ist korrekt, da der Baum T entweder eine gerade Anzahl an Knoten hat, und somit ein perfektes Matching berechnet (Siehe Punkt a)) oder genau einen Knoten zu wenig hat und somit das größte Matching um eins kleiner ist, als das perfekte Matching des nächst größeren Baums.

Die Laufzeit liegt in $\mathcal{O}(E)$, da jede Kante maximal einmal zum Matching hinzugefügt und dann aus dem Baum entfernt wird.

/ 5

- c)

/ 5

Algorithmus 1: Größtes Matching in Bäumen

```
MATCHATEE( $T = (V, E)$ )  
  // Wurzeln  
   $r \leftarrow V[1]$   
  // Wenn Knoten Blatt ist, dann die Kante zum Elternknoten zurückgeben.  
  if  $\deg(r) = 1$  then  
    // Die Kante aus dem Baum entfernen.  
     $E \leftarrow E \setminus \{(r, \text{Adj}[r])\}$   
     $V \leftarrow V \setminus \{r\}$   
    return  $\{(r, \text{Adj}[r])\}$   
  // Rekursiv für die Kinder aufrufen  
   $m \leftarrow \emptyset$   
  foreach  $v \in \text{Adj}[r]$  do  
     $m \leftarrow m \cup \text{MATCHATEE}(T)$   
  return  $m$ 
```

Aufgabe 2 – Hamiltonkreise

- a)
- b)
- c)
- d)
- e)
- f)

/ 1
/ 2
/ 1
/ 1
/ 1
/ 1

Aufgabe 3 – Perfekte Matchings in bipartiten Graphen

/ 5
